ELabELN

User Manual

Complete Guide for Laboratory Personnel

With Practical Examples, Use Cases, Design Rationale, and Cross-Industry Flexibility

Powered by LabLynx, Inc.

www.lablynx.com | www.elabeln.com

Version 3.0 | June 2026

Table of Contents

Table of Contents 2
Preface: How to Use This Manual 3
1. Introduction to ELabELN 4
1.1 What Is ELabELN? 4
1.2 The Problem ELabELN Solves 4
1.3 Cross-Industry Overview: Who Uses ELabELN? 4
1.4 Three Foundational Concepts 5
Concept 1: Experiments are the primary record. 5
Concept 2: Resources are shared, reusable records. 5
Concept 3: Teams control what is visible. 6
1.5 User Roles and What They Control 6
2. Getting Started 8
2.1 Logging In 8
2.2 The Navigation Bar 8
2.3 Your Account Settings 9
General Tab — Preferences and Display 9
Account Tab — Security and Authentication 9
API Keys Tab — Programmatic Access 10
3. Working with Experiments 11
3.1 Browsing and Filtering the Experiment Index 11
3.2 Creating an Experiment 11
3.3 Understanding Every Experiment Field 12
Title 12
Date (Started On) 13
Custom ID 13
Category 13
Status 14
Tags 15
Permissions 15
3.4 The Main Text Editor — Capturing the Science 17
Writing and Formatting 17
Tables 17
Mathematical Expressions — LaTeX 17
Code Blocks 18
Images 18
3.5 Custom Metadata Fields — Structured Data Beyond the Body 18
All Available Field Types and Their Scientific Use Cases 19
3.6 Steps — Protocol Checklists with Live Timestamps 20
3.7 Linking Experiments and Resources — Building a Knowledge Graph 21
3.8 Attaching Files — Preserving Raw Data 22

3.9 The Experiment Toolbar — All Actions Explained 22
3.10 Comments — Preserved Dialogue on the Record 23
4. Working with Resources — The Shared Inventory 25
4.1 Resource Categories — Unlimited Flexible Classification 25
4.2 Creating and Editing Resources 25
4.3 The Scheduler — Equipment Booking 26
4.4 Procurement — Reorder Management 27
5. Templates — Encoding Laboratory Standards 28
5.1 What a Template Contains 28
5.2 Creating and Managing Templates 28
5.3 Locked Steps — SOP Enforcement in Practice 29
5.4 Modular Template Composition 30
6. Search, Tags, and Organization 31
6.1 Full-Text Search 31
6.2 Tags — Multi-Dimensional Classification 31
6.3 Favorite Tags — Persistent Shortcuts 32
6.4 Four Strategies for Grouping Experiments into Projects 32
7. Signatures, Timestamps, and Record Integrity 34
7.1 What Each Tool Proves 34
7.2 Electronic Signatures — Three Levels for Different Needs 34
Level 1: Simple (Implicit) Signature 34
Level 2: Handwritten Signature Block (PDF) 34
Level 3: Advanced Cryptographic Signature (Ed25519) 35
7.3 RFC 3161 Trusted Timestamping 35
7.4 When to Timestamp 36
8. Audit Trail and Data Integrity 37
8.1 Changelog — Field-Level Change History 37
8.2 Revisions — Main Text Version History 37
8.3 ELabID — The Permanent Unique Identifier 38
8.4 Locking, Archiving, and the Deletion Model 38
8.5 ALCOA+ Data Integrity Framework — How ELabELN Satisfies Each Principle 39
9. Collaboration and Sharing 41
9.1 Sharing Within the Team 41
9.2 Request Action — Structured Review and Approval Workflows 41
9.3 Cross-Team and External Collaboration 42
9.4 Transferring Ownership 42
10. Import, Export, and the REST API 44
10.1 Exporting Experiments — Four Formats for Different Purposes 44
10.2 Importing Resources in Bulk 44
10.3 The REST API — Connecting ELabELN to the Laboratory Ecosystem 45
11. Compounds, Containers, and Special Features 47
11.1 Chemical Compounds Database 47
11.2 Containers — Hierarchical Physical Storage Tracking 47

11.3 QR Code Labels 48
11.4 The Draw Tool and JSON Editor 48
12. Configuration Patterns by Laboratory Type 50
12.1 Academic Research Lab 50
Goals: Reproducibility, collaboration, grant compliance, IP documentation, publication support, student training. 50
12.2 Pharmaceutical / Biotech Quality Assurance Lab 50
Goals: 21 CFR Part 11 compliance, CAPA documentation, method validation, batch release testing, audit readiness. 50
12.3 Forensic / Crime Laboratory 51
Goals: Chain of custody integrity, evidence traceability, court-ready records, analyst competency documentation. 51
12.4 Neurotechnology / Medical Device R&D Startup 52
Goals: Rapid deployment, IP protection, multi-team coordination, regulatory readiness, investor documentation. 52
12.5 ISO 17025 Analytical Testing / Quality Laboratory 52
Goals: Traceable method execution, equipment management, QC documentation, proficiency testing records, audit readiness. 52
13. Tips, Best Practices, and Common Pitfalls 54
13.1 Daily Use Habits That Make a Difference 54
13.2 Common Pitfalls and How to Avoid Them 54
13.3 Five Template Design Principles 55
14. Support and Resources 57

Preface: How to Use This Manual

This manual covers ELabELN from the perspective of the people who use it every day: researchers, analysts, lab managers, quality staff, and administrators. It is structured to be read sequentially during onboarding or used as a reference by feature. Every significant feature is explained at three levels:

- What it is and how to use it — the mechanics of the feature.
- Why it works this way — the design decision behind the feature and the problem it solves.
- How it is flexible — concrete examples of how the same feature serves different types of laboratories.

Four callout box types appear throughout this manual:

Tip
Orange boxes marked Tip offer shortcuts, habits, and practical guidance to help you work faster or more accurately.

Note
Navy boxes marked Note highlight important limitations, behaviors you might not expect, or configuration requirements.

Example: Green boxes marked Example
Green callout boxes show concrete, real-world scenarios. They are written from the perspective of a specific lab type to make abstract features tangible.

Use-Case Flexibility
Purple boxes marked Use-Case Flexibility summarize the range of ways a feature can be configured or applied across different laboratory types and industries.

1. Introduction to ELabELN

1.1 What Is ELabELN?

ELabELN is a cloud-hosted, browser-based Electronic Laboratory Notebook designed for scientific organizations of all types and sizes. It replaces physical paper notebooks and disconnected file systems with a single, searchable, audited digital environment where experiments, protocols, raw data, and resources are recorded, linked, and preserved.

ELabELN is built on eLabFTW, a widely adopted open-source ELN platform, and delivered by LabLynx as a fully managed subscription service. LabLynx handles hosting, backups, security patching, and user support so your team can focus on science rather than infrastructure.

The result is a system that feels like a well-organized notebook but behaves like enterprise software: it time-stamps every action, tracks every change, enforces permissions, and produces audit-ready records on demand.

1.2 The Problem ELabELN Solves

Before describing what ELabELN is, it helps to understand what it replaces. Most laboratories that transition to ELabELN describe one or more of these problems:

- Paper notebooks: Entries are undated or inconsistently dated. Handwriting is illegible to others. Notebooks are lost, damaged, or left behind when a researcher leaves. Critical reagent details are missing or cross-referenced so informally that they cannot be found later.
- Shared drives: Files accumulate with no standard naming convention. Raw data and final reports are mixed together. There is no record of who made changes or when. Instrument data files are separated from the experimental context that explains them.
- Disconnected tools: Protocols live in one place, data in another, reagent inventories in a third. Researchers spend time manually copying information between systems, introducing transcription errors. When an auditor or reviewer asks for the complete record of an experiment, assembling it takes days.
- No collaboration structure: Sharing work requires emailing files. There is no formal review or sign-off process. When a project transfers between researchers, institutional knowledge walks out the door.

ELabELN addresses each of these problems through structured data entry, automatic timestamping, searchable metadata, resource linking, configurable permissions, and formal review workflows — all within a single platform.

1.3 Cross-Industry Overview: Who Uses ELabELN?

ELabELN is designed with enough flexibility to adapt to the documentation culture of any scientific domain. The following table illustrates how the same platform serves fundamentally different laboratory environments.

Lab Type	Primary Goals	How ELabELN Helps
Academic Research	Publication, reproducibility, student training, grant compliance	Flexible templates for diverse experiment types, unlimited tags for project tracking, PDF export

		for manuscript appendices, RFC 3161 timestamps for IP protection
Pharma / Biotech QA	21 CFR Part 11 compliance, CAPA, method validation, batch records	Electronic signatures, cryptographic audit trail, locked steps for SOP enforcement, structured review/sign workflows
Forensic / Crime Lab	Chain of custody, evidence traceability, court-ready records	Custom ID fields for case numbers, strict permission controls, complete audit log, signature-on-completion workflows
Medical Device R&D	Design controls, IP documentation, multi-team coordination	Bidirectional linking, team-scoped permissions, Ed25519 signatures for invention records, API integration with PLM systems
ISO 17025 Testing	Traceable method execution, QC documentation, equipment management	Equipment scheduler with booking history, resource metadata for calibration dates, locked protocol steps, structured approval chains
Environmental / Field	Sample tracking, chain of custody, multi-site coordination	QR code labels, container/storage hierarchy, custom ID for sample barcodes, cross-team sharing for field and lab staff
Neurotechnology / Startup	Rapid deployment, IP protection, investor-ready documentation	Quick templates, RFC 3161 timestamps for prior art, REST API for instrument integration, scalable from 3 to 300 users

1.4 Three Foundational Concepts

Understanding how ELabELN is organized makes every other feature easier to learn. Three concepts underpin the entire system:

Concept 1: Experiments are the primary record.

An Experiment in ELabELN is the unit of documentation. It captures what was done, when, by whom, with which materials, following which protocol, and with what outcome. Every other object — resources, templates, tags, timestamps, signatures — exists to enrich, organize, or authenticate experiments.

> **Why it works this way**
>
> *Paper notebooks conflate protocol, execution, and result in a single undifferentiated block of text. ELabELN separates these into structured fields, attached files, linked resources, and body text, making each component independently searchable and verifiable.*

Concept 2: Resources are shared, reusable records.

A Resource represents anything a researcher uses: a reagent, an antibody, a piece of equipment, a cell line, a software tool, a reference standard. Resources are created once and linked to as many experiments as needed. When a resource record is updated — for example, when a reagent lot changes — every experiment that linked to it can be traced.

Why it works this way

In a paper notebook, reagent details are copied by hand into each experiment entry. This creates transcription errors and no connection between experiments that used the same lot. ELabELN's resource model eliminates redundancy and makes impact analysis possible: if you need to find every experiment that used a specific antibody lot, one search returns the complete list.

Concept 3: Teams control what is visible.

ELabELN organizes users into Teams. Each team has its own experiment and resource library. A user can belong to multiple teams, and experiments can be shared across teams with configurable read or write access. This makes ELabELN scalable from a single-investigator lab to a multi-division organization with dozens of teams working on separate projects.

Why it works this way

A single global experiment library would expose every researcher's unpublished work to everyone in the organization — inappropriate for competitive research environments, CROs with multiple clients, or organizations with compartmentalized compliance boundaries. Teams provide natural data boundaries without requiring separate system deployments.

1.5 User Roles and What They Control

Every user in ELabELN has a system-wide role and may have a team-level role. These interact to determine what each person can see and do.

Role	Capabilities
User (default)	Can create, edit, and delete their own experiments and resources. Can view shared entries with appropriate permissions. Cannot modify system configuration.
Admin	Has all User permissions plus the ability to manage team membership, configure templates and categories, assign tags, manage the scheduler, and view all team entries regardless of individual permissions.
SysAdmin	Has all Admin permissions plus system-level configuration: user account management, group/team creation, instance settings, LDAP/SSO integration, and API key oversight.
Locked	Read-only access. The user can view entries they have permission to see but cannot create, edit, or delete anything. Useful for auditors, external reviewers, or archived accounts.

Use-Case Flexibility

Academic lab: Most users are the default User role. The PI is Admin, allowing them to configure the team's categories and templates, view all team experiments, and manage student onboarding.
Pharma QA: QA analysts are Users; their supervisor is Admin and executes the formal sign workflow. External auditors are granted Locked accounts during the audit window, then deactivated or maintained as permanent read-only users.

ISO 17025: The quality manager is Admin and owns all templates and categories. Instrument operators are Users who execute documented methods. The Admin reviews and signs completed test records.

2. Getting Started

2.1 Logging In

Navigate to your organization's ELabELN URL (typically yourorganization.elabeln.com). You will be presented with one of two login experiences depending on how your administrator has configured authentication:

- Local credentials: Enter the email address and password associated with your ELabELN account.
- Single Sign-On (SSO): Click the SSO button and authenticate with your organization's identity provider (Microsoft Entra ID, Okta, Google Workspace, SAML 2.0, or LDAP-compatible directory). You will not maintain a separate ELabELN password.

Why it works this way

SSO is strongly recommended for organizations that already manage identities centrally. When a researcher leaves and their Active Directory account is deactivated, their ELabELN access is immediately revoked without any additional step. This satisfies the user access control requirements of 21 CFR Part 11 and ISO 17025 without manual housekeeping.

Note

If you have forgotten your local password, use the 'Forgot password?' link on the login page. If your organization uses SSO, contact your IT department — ELabELN cannot reset credentials managed by your identity provider.

Tip

Bookmark your organization's specific ELabELN URL rather than the generic elabeln.com homepage. This takes you directly to your team's login page and saves time every day.

2.2 The Navigation Bar

After logging in, a navigation bar appears at the top of every page. It is the consistent anchor point for all navigation in ELabELN. Understanding each element makes everything else faster.

Element	What It Does
Home icon / Logo	Returns you to your personal experiment dashboard — your default view of all experiments you own or can access.
Experiments	Opens the full experiment index. By default this shows your own experiments. Use the Team dropdown to switch to viewing your team's shared experiments.
Templates	Opens the template library. Templates are your team's pre-built experiment formats. Selecting a template creates a new experiment pre-populated with its structure.

Resources	Opens the resource index organized by category. This is where reagents, equipment, cell lines, and all other resource types live.
Team dropdown	Switches your active scope between teams if you belong to more than one. The scope controls which experiments and resources you see in the main index.
Search bar	Full-text search across all experiments and resources your current scope gives you access to. Searches titles, body text, metadata fields, tags, and comments.
Notification bell	Alerts you to new comments on your entries, Request Action notifications from colleagues, and system announcements.
User avatar (top right)	Opens your account settings, team settings (if Admin), API key management, and the logout option.

> **Why it works this way**
>
> *The navigation bar never changes position or content regardless of what you are doing. This is a deliberate design choice: in a complex research workflow where you might be switching between an experiment, a linked resource, and a template within minutes, a stable navigation anchor prevents disorientation and reduces the time spent thinking about the interface rather than the science.*

2.3 Your Account Settings

Click your user avatar in the top right corner and select Account Settings to access three configuration tabs that shape your personal experience of ELabELN.

General Tab — Preferences and Display

The General tab controls your display name, language, timezone, and interface theme (light or dark mode). Setting your timezone correctly is important: all timestamps in ELabELN — for entries, comments, steps, signatures, and audit events — are stored in UTC but displayed in your local timezone. An incorrect timezone makes timestamps misleading.

You can also configure your default text editor here. ELabELN offers a rich visual editor (WYSIWYG mode, similar to a word processor) and a markdown/code mode for users who prefer plain-text input with formatting syntax. You can switch modes on any individual entry regardless of the default.

> **Tip**
>
> The dark theme is popular with researchers who work in dimly lit microscopy or instrument rooms. It is not merely cosmetic — it reduces eye strain during long data entry sessions.

Account Tab — Security and Authentication

The Account tab is where you change your local password (if your organization uses local authentication) and enroll in Multi-Factor Authentication (MFA). MFA requires you to provide a time-based one-time code from an authenticator app (Google Authenticator, Authy, Microsoft Authenticator, or any TOTP-compatible app) in addition to your password.

MFA enrollment generates a QR code that you scan with your authenticator app. Store your backup codes in a secure location; these are the only way to regain access if your MFA device is lost.

Why it works this way

21 CFR Part 11 Section 11.300 requires that electronic signatures be unique to one individual and include controls to prevent their use by anyone other than the genuine owner. MFA directly satisfies this requirement. For organizations subject to FDA oversight, enabling MFA is not optional once electronic signatures are in use.

Example: MFA in a pharma quality lab

A QA analyst completes a method validation study and is ready to sign the record. She enters her password, is prompted for her MFA code, opens her authenticator app, enters the six-digit code, and signs. The signature record captures her identity, the timestamp, and the fact that two-factor authentication was used. If an FDA inspector asks how the organization prevents unauthorized signature, the MFA log provides the evidence.

API Keys Tab — Programmatic Access

The API Keys tab lets you generate personal API keys for programmatic access to ELabELN's REST API. Each key can be given a label (e.g., 'Instrument Integration — LC-MS') and can be revoked independently. Keys are shown only once at generation; store them in a secure credential manager.

Tip

Create a separate API key for each integration or script. If one key is compromised or a script is decommissioned, you can revoke that specific key without disrupting other integrations.

3. Working with Experiments

The Experiment is the core documentation unit in ELabELN. Everything in the system — resources, templates, tags, timestamps, signatures — exists in relationship to experiments. This chapter covers every field and tool available in an experiment record, explains what each does, and illustrates how each serves different types of laboratory work.

3.1 Browsing and Filtering the Experiment Index

Click Experiments in the navigation bar to open the experiment index. The index displays all experiments within your current scope (your own entries, or your team's entries if you have selected a team view). Each row shows the experiment's title, category, status, tags, date, and owner.

The index supports several ways to narrow what you see:

- Filter by category: Click any category pill in the filter panel to show only experiments of that type.
- Filter by status: Show only Running, Success, Failure, or other statuses to focus on active work or find experiments needing review.
- Filter by tag: Click one or more tags to find experiments associated with a specific project, compound, technique, or any tag dimension you have defined.
- Filter by owner: See only experiments created by a specific team member — useful for supervisors reviewing a student's work.
- Sort by date, title, or status: Click any column header to re-sort.

Tip

Use the Favorites feature (star icon on any experiment) to pin your most-active ongoing experiments to the top of the index. This is especially useful during long-running projects where you return to the same 3–5 entries dozens of times per week.

Use-Case Flexibility

Academic lab with many concurrent projects: Researchers use category filtering to separate cell biology experiments from genomics work, and tag filtering to find all experiments in a specific grant project.

QA lab during an audit: The quality manager filters by status = 'Needs Review' to queue all pending records for sign-off, and by owner to review each analyst's work independently.

High-throughput analytical lab: Analysts filter by date range and category to generate a complete list of all HPLC runs in a given week for trending.

3.2 Creating an Experiment

Click the orange + New Experiment button in the top right of the experiment index. A dialog appears with two options:

- Blank experiment: Opens a new empty experiment. You define the structure entirely yourself.

- From template: Select any template defined by your team. The new experiment is pre-populated with the template's structure — including body text, custom metadata fields, and steps — and you fill in the specific details.

Why it works this way

The two-path creation flow reflects a fundamental tradeoff between flexibility and standardization. Blank experiments let researchers document novel work without fighting a prescribed structure. Template-based experiments ensure that routine procedures are executed consistently every time — the template author has already thought through what must be captured, and the researcher cannot accidentally omit a field.

Example: Choosing between blank and template

A biochemist in a protein engineering lab creates a new template for their standard SDS-PAGE gel electrophoresis procedure. The template includes all standard metadata fields (gel percentage, buffer lot, ladder used, run voltage, run time), pre-written protocol steps with checkboxes, and a placeholder table in the body for band measurements. When a team member needs to run a gel, they create from this template: the structure is there, they fill in the actuals. But when the same biochemist tries a completely new pull-down assay for the first time, they start from a blank experiment and capture the protocol as they go, later converting it into a template for future use.

After creation, the experiment opens in edit mode. The left panel shows all structured fields; the large right area is the main text editor (body). A toolbar across the top provides access to all experiment actions.

3.3 Understanding Every Experiment Field

Every field in an experiment record serves a specific purpose. The sections below explain each field, the problem it solves, and how it is used across different lab contexts.

Title

The Title is the primary identifier of the experiment. It appears in all index views, search results, and any link that references this experiment. A well-written title makes an experiment findable years later.

Tip

Write titles that will be meaningful to someone who did not create the experiment. Include the technique, the key variable, and the outcome or purpose. Compare: 'Experiment 47' vs. 'Anti-EGFR antibody binding assay — pH 6.8 vs. 7.4 comparison — 2026-06-01'. The second is findable without opening the record.

Example: Title conventions across lab types

Academic: 'ChIP-seq on H3K27me3 marks in MEF cells — hypoxia treatment 24h — Batch 3'
Pharma QA: 'Method Validation — Potency Assay — Product X — Accuracy & Precision — Analyst JR'
Forensic: 'Case 2026-0412 — Toxicology Screen — Blood Sample B'
Environmental: 'Quarterly Water Quality Panel — Site 7 (Green River North) — Q2 2026'

Date (Started On)

The Date field records when the experiment was started. ELabELN defaults it to today's date at the moment of experiment creation, but it can be changed by the owner. This field is separate from the system-recorded creation timestamp (which cannot be modified) and is intended to capture the scientific start date — which may differ from when the record was created.

Why it works this way

Distinguishing the scientific start date from the record creation date addresses a real-world scenario: a researcher who collects data at 11 PM creates the formal experiment record the next morning. The Date field captures when the work began. The immutable system timestamp records when the record was created. Both are part of the permanent record and serve different roles in an audit or review.

Note

The Date field is not locked after the experiment is signed. If you need to change it after signing, you must unlock the experiment (which creates an audit log entry), make the change, and re-sign.

Custom ID

The Custom ID field is a free-text identifier you define. It is completely separate from the ELabELN system ID (the ELabID, which is auto-assigned and cannot be changed). The Custom ID can hold any value: a LIMS accession number, a project code, a case number, a batch ID, a grant number, or any other identifier from an external system.

Why it works this way

Most laboratories operate within a broader data ecosystem. Samples arrive with barcodes from a LIMS. Experiments are associated with grant project codes. Test requests come with case numbers. The Custom ID field is the integration point: it lets you store the external identifier alongside the ELabELN record so that cross-referencing between systems is immediate and unambiguous.

Use-Case Flexibility

Forensic lab: Custom ID = case number (e.g., '2026-CR-00412'). All experiments related to a case share the same Custom ID. Filtering by this ID instantly shows the complete analytical record for that case.

Biotech QA: Custom ID = batch number. Each batch's testing records carry the batch ID, enabling instant traceability from the test result back to the production batch.

Academic: Custom ID = grant project code (e.g., 'R01-GM-2024-003'). Filtering by grant code at reporting time produces a complete list of all experiments billable to that grant.

Environmental testing: Custom ID = sample barcode. Technicians scan barcodes from incoming samples into this field, linking physical and digital records without manual transcription.

Category

Categories are administrator-defined labels that classify experiment types. Unlike tags (which are flexible and user-created), categories are controlled vocabulary: only Admins can create, rename, or delete categories. Each experiment belongs to exactly one category.

Why it works this way

The distinction between categories (controlled, structural) and tags (flexible, user-created) is fundamental to how ELabELN organizes knowledge. Categories define the types of experiments your laboratory performs. They appear as columns in the index, drive template assignment, and enable category-level filtering. Tags capture the variable dimensions of what an experiment is about. A category is 'what kind of experiment is this?' A tag is 'what does this experiment concern?'

Example: Category design across lab types

Academic biochemistry lab: Categories = 'Protein Expression', 'Purification', 'Binding Assay', 'Crystallography', 'Cell Culture', 'Microscopy', 'Literature Review / Protocol Design'

Pharma QA: Categories = 'Method Validation', 'Stability Study', 'Release Testing', 'In-Process Control', 'CAPA Investigation', 'Reference Standard Qualification'

Forensic lab: Categories = 'Drug Chemistry', 'DNA Analysis', 'Toxicology', 'Trace Evidence', 'Firearms', 'Digital Evidence', 'QC / Proficiency'

ISO 17025 testing lab: Categories = 'Routine Analysis', 'Method Development', 'QC Check', 'Equipment Calibration', 'Proficiency Testing', 'Corrective Action'

Status

Status is a configurable state field that tracks where an experiment is in its lifecycle. Administrators define the available statuses; ELabELN ships with a default set that most labs start with.

Default Status	Meaning
Running	Experiment is in progress. Data collection is ongoing.
Success	Experiment completed with valid results.
Failure	Experiment did not produce valid results. The failure itself is scientifically important to document.
Need to be Redone	Experiment was completed but results are inconclusive or suspect. Needs to be repeated.
Waiting for Something	Experiment is paused pending an external input: a reagent delivery, an instrument repair, a colleague’s sample preparation.

Administrators can add custom statuses, rename default ones, or remove those that do not fit the laboratory’s workflow. Common custom statuses include 'Pending Review', 'Signed', 'Archived', 'On Hold — QC', and 'Approved for Release'.

Why it works this way

Failure is an important status, not an embarrassing one. In a well-run scientific record, a failed experiment documented with the reason for failure is as valuable as a success. Troubleshooting depends on this history. A lab where every experiment shows 'Success' is either extraordinarily lucky or not documenting honestly — neither of which instills confidence in reviewers or auditors.

Use-Case Flexibility

QA lab: Custom statuses 'Pending Supervisor Review' and 'Approved — Released' create a formal lifecycle between data entry and official sign-off.

Academic: Simple Running / Success / Failure workflow. The PI reviews the index filtered by 'Success' when compiling data for a manuscript.

Forensic: Custom status 'Reviewed by Supervisor' documents the technical review step that precedes report issuance.

Tags

Tags are user-defined labels that can be applied to any experiment. Unlike categories (one per experiment, admin-controlled), an experiment can have unlimited tags, and any user can create new ones. Tags are searchable, filterable, and can be visualized as a network graph showing relationships between experiments.

Why it works this way

Tags are the flexible classification layer. Because science rarely fits neatly into a single hierarchy, experiments need to be classifiable along multiple independent dimensions simultaneously. A single experiment might be tagged with the project name, the compound tested, the collaborator's initials, the technique used, and the publication it supports — and it should be findable by any of these dimensions independently.

Example: Multi-dimensional tagging in practice

A researcher studying a novel kinase inhibitor tags each experiment with: (1) the compound identifier ['Compound-17B'], (2) the assay type ['HTRF', 'Western Blot', or 'Cell Viability'], (3) the cell line used ['HeLa', 'A549'], and (4) the project code ['Proj-KI-2025']. Three months later, when a reviewer asks for all HeLa cell data for Compound-17B, the researcher filters by both tags simultaneously. The complete list of matching experiments appears in seconds.

Tip

Develop a team-wide tag naming convention before data accumulates. Decide on capitalization, whether to use spaces or hyphens, and which dimensions you want to tag consistently (project, compound, technique, collaborator). A tag convention created retroactively when hundreds of experiments already exist is painful to apply.

Permissions

Every experiment has an explicit permission setting that controls who can view and edit it. The owner always has full access. Beyond the owner, four levels are available:

Permission Level	Access Granted
Private (owner only)	Only you can see or edit this experiment. No one else on your team has any access, regardless of their role.
User (team members)	All members of your team can view the experiment. Only you (and Admins) can edit it.
Team (read/write)	All team members can view and edit the experiment. Use for truly collaborative entries where multiple people contribute.

Organization	All teams in the organization can view this experiment. Useful for cross-departmental reference records, shared SOPs, or institutional templates.

Why it works this way

Default privacy protects unpublished work. A researcher who has not yet published should not have their experiment data visible to everyone in the organization by default. The opt-in sharing model means researchers actively choose when their work is ready to be seen, which protects IP, prevents premature disclosure of results, and respects the competitive nature of scientific research.

Use-Case Flexibility

Academic: Students set experiments to 'User' level so their PI can review progress without the whole lab seeing draft data. When the work is ready to submit, they may change to 'Organization' if the institution uses ELabELN as a shared repository.

CRO (Contract Research Org): Each client project is a separate team. Experiments are kept at 'Team' level within the client's team. The sponsoring client's liaison has read access; no other client can see the data.

Pharma: All executed QA records are set to 'User' level during execution. The supervisor's Admin role lets them view all records regardless. After approval, status is changed and the record is locked.

3.4 The Main Text Editor — Capturing the Science

The main text editor is the body of the experiment record — the equivalent of the blank page in a paper notebook, but with structured capabilities. It is where researchers write protocols, record observations, paste or type results, construct data tables, and embed images.

Writing and Formatting

The editor supports headings, bold, italic, underline, strikethrough, superscript, and subscript formatting. Organize long entries using heading levels to create navigable sections: a heading for 'Protocol', another for 'Results', another for 'Analysis', and another for 'Conclusions'. The entry's internal structure becomes visible in the document outline and makes long entries navigable without scrolling.

> **Tip**
>
> Use heading structure deliberately. A consistent heading scheme across all experiments in a category makes the records more useful to reviewers and makes future retrieval faster. If your team uses 'Objective / Materials / Protocol / Results / Discussion' in every experiment of a given type, that structure becomes implicit quality documentation.

Tables

ELabELN's built-in table editor lets you create tables with configurable rows and columns directly in the experiment body. Tables are sortable: clicking any column header sorts the table by that column, ascending or descending. This is powerful for result sets where you want to rank compounds by activity, sort samples by result value, or find outliers in a data set.

> **Example: Sortable result table in an analytical lab**
>
> An HPLC analyst runs 24 samples in a single batch. Each sample's result (retention time, area, concentration) is entered into a table in the experiment body. After entry, the analyst clicks the Concentration column header to sort ascending — immediately identifying the three samples below the specification limit. They add a note to those rows and flag the experiment status as 'Needs Review'. The sort operation takes two seconds. Finding the same information in a flat text document would require manual scanning.

Mathematical Expressions — LaTeX

ELabELN supports LaTeX syntax for mathematical notation. Surround a LaTeX expression with dollar signs ($...$) for inline math or double dollar signs ($$...$$) for display math. The expression renders visually in the saved entry.

- Inline example: $R^2 = 0.9987$ renders as a properly typeset R-squared value inline in text.
- Display example: $$C = \frac{A}{\varepsilon \cdot l}$$ renders the Beer-Lambert law as a centered, properly formatted equation.

> **Why it works this way**
>
> *Scientific documentation has always required mathematical notation, and the workaround in most digital environments — inserting screenshots of equations from another tool — creates images that are not searchable, not renderable at different scales, and not accessible to screen readers. LaTeX renders the equation as a first-class element of the document.*

Code Blocks

Insert a code block via the toolbar (the {} icon) to include programming code, command-line instructions, or data processing scripts with syntax highlighting and monospace formatting. Code blocks preserve whitespace and indentation, making Python, R, bash, or any other language readable within the experiment record.

Example: Code block in a bioinformatics experiment

A computational biologist documents a RNA-seq alignment step. In the experiment body, they insert a code block containing the exact STAR aligner command with all parameters: genome directory, read files, output location, thread count, and mismatch tolerance. The code block is preserved exactly as run. Six months later, when replicating the analysis, any team member can copy the command directly from the experiment record without retyping or risk of parameter drift.

Images

Images can be inserted directly into the body text by drag-and-drop, copy-paste from the clipboard, or the image insertion tool in the toolbar. Images are stored as part of the experiment record and rendered inline at a configurable display size. You can add captions and resize the display without affecting the original file.

Why it works this way

Embedding images in the body — rather than attaching them as separate files — preserves the narrative relationship between the image and the text that explains it. A gel image appears immediately below the protocol step that produced it. A microscopy image appears next to the quantification notes. The record reads as an integrated scientific account rather than a document accompanied by a folder of unlabeled images.

Tip

For instrument images (gels, western blots, micrographs), paste the image directly into the body at the point in the protocol where it was taken, then type your annotation immediately below it. This practice produces experiment records that are self-contained and readable without the original file context.

3.5 Custom Metadata Fields — Structured Data Beyond the Body

Custom metadata fields are structured data fields that appear below the title/date/category block and above the body. They capture specific, defined pieces of information in a machine-readable format — not as unstructured text, but as typed values that can be searched, filtered, validated, and exported.

Metadata fields are defined in templates by Admins. When an experiment is created from a template, that template's fields appear automatically. They can also be added manually to any experiment by selecting from the team's field library.

Why it works this way

Free-text body content is human-readable but machine-opaque. A researcher who writes 'temperature: 37 degrees C' in the body has captured the information, but a query asking 'find all experiments run at 37°C' cannot parse it reliably. A metadata field of type Number labeled

'Incubation Temperature (°C)' with the value 37 is queryable, filterable, and exportable. Metadata fields convert the body's narrative into structured data.

All Available Field Types and Their Scientific Use Cases

Field Type	Format	Scientific Use Cases
Text	A single line of free text	Buffer name, lot number, instrument ID, operator initials, external reference number
Text area	Multiple lines of free text	Sample description, deviation notes, observation summary, non-structured comments that belong in metadata rather than the body
Number	A numeric value with optional units	Concentration (mg/mL), volume (µL), temperature (°C), pH, molecular weight, yield (%), assay incubation time (min)
Date	A calendar date picker	Sample receipt date, expiration date, calibration due date, test request date, protocol version date
Date and time	Date plus hours/minutes	Instrument run start time, autoclave cycle timestamp, centrifugation start/end for chain of custody
Checkbox	True / False toggle	GMP-compliant run? / Duplicate performed? / Blank subtracted? / Approved for release?
Select	Single choice from a defined list	Instrument used (from list of calibrated instruments), analysis method (from validated methods), analyst (from trained personnel list)
Multi-select	Multiple choices from a list	Tags at the metadata level (compounds tested, techniques used, matrix types); multiple co-investigators; concurrent analysis modes
Radio button	Single choice with visual radio format	Pass/Fail/Inconclusive; Positive/Negative/Equivocal; Compliant/Non-Compliant/Borderline
URL	A hyperlink	Link to instrument raw data in a connected repository, instrument manufacturer's SOP online, regulatory guidance document, related project page in PM system

Example: Metadata fields for an ELISA in a pharma QA lab

Text: 'Analyst Name' — value: 'J. Rodriguez'
Text: 'Plate Lot Number' — value: 'PLT-2026-0055'
Select: 'Instrument' — value: 'Victor Nivo #3 (Cal. 2026-05-15)'
Number: 'Plate Incubation Temperature (°C)' — value: '37'
Number: 'Sample Dilution Factor' — value: '100'
Date: 'Kit Expiration Date' — value: '2026-12-31'
Checkbox: 'Blank Subtraction Applied?' — value: checked

Radio: 'Result Assessment' — value: 'Pass'

Use-Case Flexibility

Genomics lab: Metadata fields capture read depth (Number), reference genome version (Select), aligner version (Text), and whether the analysis was re-run after QC failure (Checkbox).

Environmental testing: Fields capture collection site (Select from validated site list), collection time (Date and time), field technician (Select from trained personnel), and preservation method (Select).

Neurotechnology startup: Metadata fields for device model (Text), firmware version (Text), implantation site (Select), and recording duration (Number) make every neural recording experiment queryable by hardware configuration.

3.6 Steps — Protocol Checklists with Live Timestamps

Steps are a checklist panel attached to an experiment. Each step is a protocol instruction that the researcher checks off as they complete it. When a step is checked, ELabELN records the exact date, time, and identity of the user who checked it. This transforms the protocol from a plan into a live execution record.

Steps can be added manually or defined in a template. In templates, steps can be locked (see Section 5.3 for details). Each step also has a body text area for notes, enabling the researcher to add execution notes, deviation observations, or measurements at the step level rather than the experiment level.

Why it works this way

Paper SOPs are checked off with a pen, providing no information about when each step occurred, who performed it, or whether the timing between steps was correct. ELabELN's step timestamps create a time-resolution record of protocol execution. For time-sensitive procedures — where step 3 must be performed within 5 minutes of step 2 — the timestamps provide evidence that the procedure was followed correctly. For GMP purposes, this is the difference between a record that says 'I followed the protocol' and one that proves it.

Example: Steps in a forensic drug chemistry analysis

Step 1: 'Retrieve evidence sample from secure storage' — timestamped at 09:14 by Analyst Kovacs

Step 2: 'Weigh sample and record net weight' — timestamped at 09:18 by Analyst Kovacs

Step 3: 'Prepare color test reagents per SOP-DC-003' — timestamped at 09:22 by Analyst Kovacs

Step 4: 'Perform Scott test' — timestamped at 09:25 by Analyst Kovacs; note: 'Blue color change observed within 30 seconds'

Step 5: 'Prepare GC-MS sample per SOP-DC-004' — timestamped at 09:31 by Analyst Kovacs

Step 6: 'Return evidence to secure storage with chain of custody entry' — timestamped at 09:44 by Analyst Kovacs

Result: The entire evidence handling chain is documented with minute-level precision, entirely within the ELabELN record.

Use-Case Flexibility

Academic: Steps serve as a protocol reminder during complex multi-step procedures. The researcher checks off steps as they go, and the timestamps confirm the timing of each step in the experiment record.
Biotech GMP: Locked steps from a validated template cannot be reordered, skipped, or modified. The step completion record proves the validated protocol was followed exactly.
ISO 17025: Steps document the execution of a standard method. Any deviation from the expected step sequence — visible from the timestamps — triggers a deviation note.

3.7 Linking Experiments and Resources — Building a Knowledge Graph

ELabELN allows any experiment to be linked to any other experiment or resource. Links are bidirectional: if Experiment A links to Resource B, Resource B's record also shows that Experiment A linked to it. This creates a navigable network of related records rather than isolated documents.

To create a link, click the Links panel in the experiment sidebar (edit mode) and type to search for the target entry by title, ELabID, or Custom ID. Multiple links can be created from one entry.

Why it works this way

Scientific knowledge is not linear. An experiment builds on previous experiments. A protocol references multiple reagents, each with its own record. A compound's properties span dozens of experiments. In a paper system, the only way to represent these relationships is by hand-writing cross-references that are error-prone and quickly become stale. ELabELN's linking model makes these relationships first-class, permanent, and navigable. Clicking a link on an experiment instantly opens the linked resource or experiment in context.

Example: Linking in a neurotechnology startup

A researcher records a neural recording experiment and links it to: (1) the Probe Resource record for the specific electrode array used, including its batch number and impedance specifications; (2) the Animal Resource record for the subject, including surgical notes and housing conditions; (3) the previous experiment where the animal was surgically implanted; and (4) the analysis script Resource record documenting the spike-sorting parameters.
When the team needs to investigate a data quality issue six months later, they open the recording experiment and navigate directly to all linked records in two clicks. The complete experimental context — which would have required searching four separate folders in a shared drive — is assembled instantly.

Tip

For ongoing projects, establish a 'Project Summary' experiment at the start of the project and link every subsequent experiment in the project to it. The Project Summary becomes a navigable table of contents for the entire body of work, and every experiment in the project is one click away from any other.

Use-Case Flexibility

Pharma: Link each CAPA investigation experiment to the original OOS (Out-of-Specification) result experiment. The investigation record inherits the context of the trigger event.

Academic: Link all experiments that contribute data to a specific manuscript figure. When reviewers ask for underlying data, the figure-linked experiments are immediately available.

Environmental: Link each field collection experiment to the laboratory analysis experiment for the same sample. The sample record is complete and traceable from collection through result.

3.8 Attaching Files — Preserving Raw Data

The Attachments panel (below the main body in view mode, accessible as a drag-drop target in edit mode) holds any files associated with the experiment. There is no limit on the number or type of files: instrument raw data files, chromatograms, spectra, gel images, plate reader outputs, analysis scripts, PDFs, videos, and any other format are all accepted.

Each file attachment shows its filename, size, upload timestamp, and uploader identity. Files can be downloaded individually or bulk-downloaded as a ZIP archive.

Why it works this way

The most common data integrity gap in paper-based or shared-drive systems is the separation of raw data from the record that explains it. Instrument files accumulate in shared drives named by date or run number, with no connection to the experimental context that generated them. When an auditor or reviewer asks 'show me the raw chromatogram for this result,' the answer often involves an extensive manual search. ELabELN's attachment model makes the raw data and its experimental context inseparable: the chromatogram is part of the experiment record, not a file somewhere in a folder.

Example: Attachment practice in an analytical testing lab

An HPLC analyst runs a potency assay. After the run, they drag the instrument's raw .raw file, the processed peak integration report (.pdf), and the sequence file (.seq) directly onto the experiment record. All three files are now permanently associated with that analysis event, timestamped at the moment of upload, and identified by the analyst's account.

When the QA manager conducts the review, she opens the experiment, clicks the chromatogram PDF in the attachments panel, and reviews the integration directly in the browser. No emails requesting files. No shared drive navigation. The complete record is in one place.

Tip

Always attach the original raw instrument output file (e.g., .raw, .wiff, .d, .cdf) alongside any processed analysis. The raw file is the source of truth. Processed reports can be regenerated from raw data; raw data cannot be reconstructed from a report. This satisfies the ALCOA+ 'Original' requirement.

3.9 The Experiment Toolbar — All Actions Explained

The toolbar at the top of an experiment record (visible in both view and edit mode) provides access to every action that can be taken on the experiment. The following table explains each action in the order it typically appears.

Action	What It Does

Edit / Save	Toggles the entry between view mode (read-only, clean display) and edit mode (all fields editable). In edit mode, changes are auto-saved periodically; click Save to commit immediately.
Duplicate	Creates a new experiment with the same structure as this one: same fields, body content, metadata, and steps. The new entry gets a new ELabID, a new creation timestamp, and an empty attachment list. Permissions and status reset to defaults.
Share / Permissions	Opens the permission dialog to change who can view or edit the experiment beyond the current permission setting.
Sign	Opens the signature dialog. Choose the signature type (simple, PDF, or Ed25519). After authentication, a signature block is appended to the experiment body and logged in the audit trail. The entry is locked after signing (can be configured).
Timestamp	Sends the experiment's cryptographic hash to an RFC 3161 timestamp authority (or blockchain) and stores the timestamp token with the record. Proves the content existed at this moment in time.
Export	Exports the experiment to PDF (for human reading and archiving), ZIP archive (all content plus raw files), or ELN format (RO-Crate, for import into another compliant ELN). The PDF export includes all metadata, body, attachments list, steps, and audit information.
Pin Entry	Pins this experiment to the top of your experiment index for quick access. Useful for high-frequency active experiments in a long-running study.
Lock / Unlock	Locks the experiment against further editing. The lock event and timestamp are logged. Only the owner or an Admin can unlock. Unlocking creates a new audit event.
Request Action	Sends a structured notification to a named colleague requesting they perform a specific action on this experiment: Archive, Lock, Review, Sign, or Timestamp.
Ellipsis (…)	Reveals less-common actions: Transfer Ownership, See Revisions (full body version history), See Changelog (field-level change history), Hide Main Text (collapses body in index view), Archive, and Delete.

Example: Duplicate in a high-throughput analytical lab

An HPLC analyst runs the same potency assay method three times per week. Rather than creating a new experiment from scratch each time, she duplicates the previous week's entry, which already has the correct metadata fields, instrument reference, and body structure pre-populated. She updates only the date, batch number, analyst note, and results. Setup time drops from 15 minutes to 90 seconds. All historical entries remain intact and individually searchable.

3.10 Comments — Preserved Dialogue on the Record

Any user with read access to an experiment can leave a comment. Comments are timestamped, attributed, threaded, and permanent. Owners and commenters receive notifications when new comments are added. Comments cannot be deleted by anyone, including Admins — they are part of the permanent record.

Why it works this way

The permanence of comments is a deliberate data integrity decision. In any regulated or legally reviewed context, the ability to delete comments would make the record manipulable: a supervisor could remove inconvenient observations, or a researcher could delete a concern they later wished they had not raised. Permanent, attributed, timestamped comments create a communication layer on the record that is as trustworthy as the record itself. This is critical in peer review, GMP review, forensic testimony, and litigation support contexts.

Example: Comments across three workflows

Manuscript review: A PI reviews an experiment and comments: 'The statistical analysis in Table 3 uses the wrong test for non-parametric data. Please redo using Mann-Whitney U.' The researcher replies two days later: 'Corrected. Updated figure attached. P-value is now 0.032.' The entire exchange is permanently associated with the experiment.

GMP deviation: During QA review, a supervisor comments: 'Incubation temperature recorded as 37°C but the incubator log shows a 40-minute excursion to 39.1°C on this date. Please investigate and document impact assessment.' The analyst acknowledges, investigates, and documents the conclusion in a follow-up comment. The CAPA chain is complete within the experiment record.

Forensic review: A technical reviewer comments on a drug chemistry analysis: 'Color test result documented as positive but intensity score not recorded per SOP-DC-003. Please amend.' The analyst adds the missing information to the body (creating a revision) and confirms in a reply comment. The reviewer’s concern and the resolution are both on the record.

4. Working with Resources — The Shared Inventory

Resources are shared records representing anything a laboratory uses: reagents, equipment, cell lines, software, reference standards, materials, instruments, organisms, and more. Resources are created once, maintained centrally, and linked to from as many experiments as needed. When a resource record is updated, every experiment that references it gains access to the current information.

4.1 Resource Categories — Unlimited Flexible Classification

Like experiments, resources are organized into administrator-defined categories. There is no limit to the number of categories, and the categorization scheme is entirely under your control. Resources can represent any physical or logical entity your laboratory tracks.

Use-Case Flexibility

Academic biochemistry: Antibodies, Plasmids, Cell Lines, Bacterial Strains, Primers, Enzymes, Chemicals, Instruments, Software Tools, Published Protocols

Pharma/Biotech QA: Reference Standards, Reagent Kits, Instruments, Validated Methods, Trained Personnel, SOPs (as resource records), Supplier Qualification Records

Forensic lab: Evidence Items, Certified Reference Materials, Instruments, QC Materials, Reagents, Personnel (certification tracking), Proficiency Test Records

Neurotechnology startup: Electrode Arrays, Animal Subjects, Surgical Instruments, Analysis Scripts, Device Firmware Versions, Implantation Protocols

Environmental testing: Certified Reference Standards, Field Equipment, Preservation Reagents, Sampling Containers, Accredited Methods, Field Technicians

Why it works this way

The category system's unlimited flexibility means ELabELN's resource module can serve as the laboratory's inventory, equipment register, personnel competency log, reference material catalog, and SOP library simultaneously — all within a single searchable, linked system. The category boundary is defined entirely by what the lab decides to track, not by what the software prescribes.

4.2 Creating and Editing Resources

Click Resources in the navigation bar, select a category, and click + New Resource. The resource form mirrors the experiment form: a title, date, category, custom metadata fields, a body text area, and an attachment panel. The same field types available for experiments (Text, Number, Date, Select, Checkbox, etc.) are available for resources.

Example: A complete antibody resource record

Title: 'Anti-Phospho-ERK1/2 (Thr202/Tyr204) mAb, Clone D13.14.4E, Cell Signaling Technology #4370'

Custom Metadata: Catalog Number (Text) = '4370S'; Lot Number (Text) = 'LOT 0012'; Host Species (Select) = 'Rabbit'; Application (Multi-select) = 'WB, IF, IHC'; Dilution WB (Text) = '1:1000'; Dilution

IF (Text) = '1:200'; Storage Temp (Select) = '-20°C'; Expiration Date (Date) = '2027-08-15'; Location (Text) = 'Freezer 3, Box 12, Slot C4'

Body text: Protocol notes for stripping and reprobing, observations about background on specific cell lines, performance notes added over time by multiple users

Attachments: Manufacturer's datasheet PDF, validation western blot image

Linked experiments: All 23 experiments where this antibody has been used link here. Any researcher can open the antibody record and see the complete usage history across all experiments, all users, and all dates.

Tip

Resource records improve over time. Every time a researcher uses a reagent and discovers something new — an unexpected cross-reactivity, an ideal dilution for a specific application, a storage note — they should add it to the resource body. The resource record becomes a living, collective knowledge document about that reagent, more valuable than the manufacturer's datasheet.

4.3 The Scheduler — Equipment Booking

Any resource can be made bookable. When an equipment resource (an instrument, a biosafety cabinet, a microscope, an autoclave, a centrifuge) has scheduling enabled, a calendar appears on its resource page. Authorized users can book time slots. The scheduler shows all bookings in a color-coded calendar view.

Feature	Details
Why enable scheduling?	Prevents double-booking of shared equipment. Creates a searchable usage log. Enables instrument utilization analysis. Provides evidence that the equipment was available and used during the documented experiment window.
Booking fields	Users specify a start time, end time, and optional note. Admins can restrict who can book specific resources and require approval for bookings.
Kiosk mode	A resource can be placed in 'kiosk mode' — a simplified, full-screen view of its calendar suitable for display on a tablet mounted next to the equipment. Users can walk up and book or check availability without navigating the full ELabELN interface.
Usage log	All bookings are permanently logged with user identity, time, and notes. This log can serve as equipment usage documentation for ISO 17025 and similar standards requiring equipment records.

Example: Kiosk mode in a neurotechnology lab

A neurotechnology startup mounts an Android tablet running ELabELN kiosk mode beside their 16-channel acquisition system. Researchers tap their name, select a time slot, and add a brief note directly at the instrument. The booking is logged in ELabELN. When the PI reviews equipment utilization at month end, the complete acquisition system usage log is already in ELabELN, timestamped and attributed, with no manual log book required.

Use-Case Flexibility

Academic core facility: All instruments are made bookable. The facility manager monitors the calendar to plan preventive maintenance during low-usage windows. Booking history provides evidence of instrument demand for funding requests.

ISO 17025 lab: The centrifuge, balance, and pH meter are bookable resources. Each resource record also contains calibration due dates. Analysts can see immediately whether the instrument's current calibration covers their booking date.

Pharma: The autoclave is a bookable resource with an approval step. Each cycle requires supervisor sign-off before the booking is confirmed, providing an authorization record for sterilization events.

4.4 Procurement — Reorder Management

Any resource can have a minimum stock quantity defined and a current stock quantity tracked. When the current quantity falls below the minimum, ELabELN flags the resource for reorder and can notify designated users. This turns the resource module into a lightweight procurement management system.

The procurement feature is intentionally simple: it is not a full purchasing system, but a trigger and notification layer. The actual purchase order process is handled in whatever system the organization uses. ELabELN's role is to ensure that the reorder signal is never missed because someone forgot to check the freezer.

Tip

Set the minimum stock quantity higher than the absolute minimum you need. Lead times for reagents can be unpredictable. A minimum that triggers reorder when you have 4 weeks of stock remaining is safer than one that triggers when you are about to run out.

5. Templates — Encoding Laboratory Standards

Templates are the mechanism by which individual laboratory practice becomes institutional standard. A template is a pre-built experiment structure: the body text, metadata fields, steps, tags, and category are all predefined. When a researcher creates an experiment from a template, they inherit the structure and fill in the specifics. Templates ensure that routine procedures are documented consistently, regardless of which researcher performs them.

5.1 What a Template Contains

A template can contain any element that an experiment can contain, set at the time of template creation:

- Pre-written body text: Protocol steps written out in prose, section headers, tables pre-formatted for result entry, placeholder images, standard disclaimers, or SOP references.
- Custom metadata fields: All the structured fields that must be captured for this experiment type. A template for an ELISA might define 10 fields; a template for a cell count might define 3.
- Steps: Pre-written protocol steps that appear as a checklist on every experiment created from this template. Can be locked (see Section 5.3).
- Category: The template auto-assigns the experiment to the correct category.
- Default status: Set the initial status automatically (e.g., all new experiments start as 'Running').
- Default tags: Apply standard tags automatically (e.g., every method validation experiment is auto-tagged 'Validation').

> **Why it works this way**
>
> *The reason templates exist is that the most expensive cognitive work in routine laboratory documentation is not filling in data — it is deciding what to document. A template externalizes that decision into a one-time design activity performed by a knowledgeable scientist. Every subsequent researcher who uses the template benefits from that decision without having to remake it. The result is consistent, complete records without additional cognitive load on the end user.*

5.2 Creating and Managing Templates

Templates are created and managed by Admins (or by Users with template creation permission granted by an Admin). Navigate to Templates in the navigation bar and click + New Template. The template editor is identical to the experiment editor: you build the template exactly as you would want an experiment of that type to look.

Best practices for template creation:

- Write body section headers first, then populate each section with placeholder text indicating what goes there. This gives the researcher structural guidance without prescribing their words.
- Define all metadata fields that must be captured. Include every field that would ever be needed for audit, review, or analysis. It is easier to leave an optional field blank than to reconstruct missing data later.
- Write steps for every procedural element of the protocol. The step list is the SOP in executable form.
- Version your templates. When a SOP is revised, create a new template version rather than modifying the existing one. Old experiments created from the previous template retain their original structure.

Example: Template for a Bradford Protein Quantification Assay

Body: [Section: Objective] placeholder text — Researcher fills in sample description and purpose.

Body: [Section: Standard Curve] pre-formatted table with BSA concentration (mg/mL) and absorbance (A595) columns, 8 rows pre-labeled.

Body: [Section: Sample Results] table with Sample ID, Dilution Factor, A595, and Calculated Concentration columns.

Body: [Section: Observations and Notes] empty section for analyst notes.

Metadata: Analyst Name (Text), Instrument (Select: BioRad SmartSpec, NanoDrop 2000, other), Reagent Lot (Text), Date of Assay (Date), Plate Reader Version (Text).

Steps: (1) Prepare BSA standards per SOP-PRO-001; (2) Prepare sample dilutions; (3) Pipette standards and samples in triplicate; (4) Add Bradford reagent; (5) Incubate 10 min at room temp; (6) Read absorbance at 595 nm; (7) Enter values into table above; (8) Calculate concentrations using standard curve equation.

Category: Protein Quantification. Status: Running. Tags: auto-tagged 'Assay' and 'QC'.

5.3 Locked Steps — SOP Enforcement in Practice

Steps in a template can be individually locked. A locked step cannot be modified, added to, reordered, or deleted by a User once the experiment is created. Only an Admin can change locked steps, and doing so requires modifying the template (which creates a new version), not the individual experiment.

Why it works this way

The distinction between locked and unlocked steps reflects the difference between SOPs and flexible protocols. A SOP step is fixed: the regulatory commitment is that this exact procedure was followed. An unlocked step is guidance: the researcher is expected to complete it but may adapt it. Locked steps make the SOP commitment machine-enforced rather than honor-system-enforced. In a GMP environment, this distinction is the difference between a system that prevents deviation and one that merely requests compliance.

Example: Locked steps in a GMP biotech lab

A validated cell-based potency assay has 12 steps in its SOP. All 12 are locked in the ELabELN template. When an analyst creates an experiment from this template, she sees the 12 steps and can check them off as completed, but she cannot change their wording, skip one, or add a new step between existing ones. If she needs to deviate from the SOP, the deviation must be documented in the experiment body and reviewed by QA — the locked step structure itself has not been violated. An FDA inspector reviewing the record can confirm that the validated procedure was what the analyst followed: the steps are exactly as validated, and the timestamps prove when each was completed.

Use-Case Flexibility

Academic: Most steps are unlocked, giving researchers flexibility to adapt published protocols. A few critical steps (e.g., 'Record animal ID before any procedure') may be locked for animal research compliance.

ISO 17025: All steps in accredited method templates are locked. Any modification to an accredited method requires a formal change control process, not an ad hoc template edit.

Forensic: Steps for evidence handling, chain of custody, and quality control are locked. Analytical steps may have some flexibility. The locked/unlocked division reflects which parts of the procedure have regulatory force.

5.4 Modular Template Composition

Complex workflows often consist of sub-procedures that are reused across multiple larger procedures. ELabELN supports a modular approach: you can create small, focused templates for individual sub-procedures and compose them by linking experiments. A large workflow is then represented as a set of linked experiments, each documented with its own focused template.

Example: Modular composition in a protein characterization workflow

Template 1: 'Expression — E. coli BL21 IPTG Induction' — covers induction conditions and cell harvest.

Template 2: 'Purification — Ni-NTA Affinity Chromatography' — covers column equilibration, loading, washing, and elution.

Template 3: 'Purification QC — SDS-PAGE' — covers gel setup, run, staining, and interpretation.

Template 4: 'Characterization — SEC-MALS' — covers instrument setup, sample injection, and molecular weight determination.

Each experiment is created from its respective template. All four are linked to a master 'Protein Batch X Production' summary experiment. The complete production record is navigable from any of the four linked experiments.

Tip

Modular templates are better than one large template when: (1) the sub-procedures can be done by different people, (2) the sub-procedures are reused across multiple different workflows, or (3) the timing between sub-procedures spans multiple days. One experiment per day per major step keeps records focused and navigable.

6. Search, Tags, and Organization

The value of a laboratory notebook is only realized if the records it contains can be found. ELabELN provides multiple complementary ways to organize and retrieve experiments: full-text search, tag-based classification, category and status filtering, and flexible project grouping strategies.

6.1 Full-Text Search

The search bar in the navigation bar performs a full-text search across all experiments and resources you have access to. The search index includes:

- Experiment and resource titles
- Body text content (including content inside tables)
- All custom metadata field values
- Tag names
- Comment text
- Attachment file names
- ELabIDs and Custom IDs

Results are returned in relevance order and can be further filtered by the standard filter panel (category, status, tag, date, owner). Search does not require exact phrase matching: ELabELN's search is designed to find partial matches and related terms.

Example: What full-text search finds

A researcher searches for 'Compound 17B cytotoxicity'. Results include: all experiments that contain the phrase 'Compound 17B' anywhere in title, body, or metadata; all experiments tagged 'Compound-17B'; all experiments in the 'Cytotoxicity' category; all resource records for Compound 17B. A single search retrieves the complete picture across experiments, resources, tags, and metadata without requiring a structured query.

Tip

Search is your fastest path to anything in ELabELN. Resist the urge to navigate by folder structure — there is no folder structure, and you do not need one. Use the search bar the way you use a search engine: describe what you are looking for in plain terms, then filter results if needed.

6.2 Tags — Multi-Dimensional Classification

Tags are described in Section 3.3 from the perspective of individual experiment configuration. From an organizational perspective, tags are the primary tool for grouping experiments across categories, dates, and owners into meaningful project-level collections.

ELabELN's tag view offers two display modes:

- List view: Tags are displayed alphabetically with the number of experiments each tag is applied to. Click any tag to see all tagged experiments.
- Graph view: Tags are displayed as a network graph where nodes are tags and edges connect tags that frequently appear together on the same experiment. The graph reveals relationships between research threads that may not be obvious from alphabetical lists. A cluster of interconnected tags may indicate a body of work ready for analysis or publication.

Why it works this way

The graph view answers a question that list views cannot: 'Which of my research themes are connected, and how?' A node with many edges is a tag that bridges multiple areas of work. A densely connected cluster of tags is a body of work that may be coalescing around a coherent narrative. This kind of structural insight is impossible in a linear notebook but emerges naturally from the ELabELN tag graph.

6.3 Favorite Tags — Persistent Shortcuts

Any tag can be marked as a favorite by clicking the star next to it in the tag list. Favorited tags appear as quick-access links at the top of the Experiments page. For researchers working on 2–3 active projects, this means one click separates all experiments in Project A from all experiments in Project B, without typing a search.

Tip

Create one favorite tag per active project. At the start of each week, click the project tag to review all experiments in that project at a glance. This habit replaces the end-of-project 'where is everything' search with a continuous, live project dashboard.

6.4 Four Strategies for Grouping Experiments into Projects

ELabELN does not have a dedicated 'project' object. This is intentional: different labs define 'project' differently, and a rigid project structure would not serve all of them. Instead, ELabELN provides four complementary tools for grouping, each appropriate for different organizational needs.

Strategy	How It Works	Strengths
Tag-based grouping	Apply a project tag to every experiment in the project	Simple, flexible, no overhead. Best for most labs. Multiple tags can be applied to one experiment (one experiment can belong to multiple projects).
Category-based grouping	Create a category per project or project type	Clear structural separation. Works best when projects have distinct experiment types.
Custom ID grouping	Populate the Custom ID field with a project code on every experiment	Enables cross-system integration. Best when the project code also lives in a LIMS, ERP, or PM system.
Linked summary experiment	Create a 'Project Summary' experiment and link all project experiments to it	Creates a navigable master index for the project. The summary becomes the project's table of contents.

Example: Combined strategy in a pharma research lab

A research team working on a kinase inhibitor program uses all four strategies in combination: (1) Every experiment is tagged 'Proj-KI-Alpha'. (2) Experiments are in categories: 'Biochemical Assay', 'Cell Biology', 'ADME', 'Chemistry'. (3) Every experiment has the project's internal code in the Custom ID field, matching the project management system. (4) A 'Project Alpha — Master Summary' experiment is maintained by the project lead and updated weekly with links to key experiments and a progress narrative. Any team member can find all project experiments via the tag, navigate by experiment type via category, cross-reference with the PM system via Custom ID, and get an instant overview via the summary experiment.

7. Signatures, Timestamps, and Record Integrity

ELabELN provides three complementary tools for establishing the authenticity, integrity, and attribution of experimental records: electronic signatures, RFC 3161 trusted timestamps, and blockchain timestamps. These tools address different questions about a record and are used independently or in combination depending on the regulatory and evidentiary requirements of your laboratory.

7.1 What Each Tool Proves

Tool	It Proves...	Primary Use Cases
Electronic Signature	This specific person reviewed and approved this record on this date	Authorship, authorization, and accountability. Required by 21 CFR Part 11 and ISO 17025 for certain record types.
RFC 3161 Timestamp	This exact content existed at this exact date and time	Prior existence and content integrity. Used for IP protection, data priority disputes, and anti-backdating proof.
Blockchain Timestamp	This exact content existed at this exact date and time (via decentralized verification)	Same as RFC 3161, with verification that does not depend on a single trusted authority.

Why it works this way

The three tools answer three different questions. A signature answers 'did this person approve this?' A timestamp answers 'when did this content exist?' A record can be timestamped without being signed (e.g., a researcher's raw experimental data that has not yet gone through review). A record can be signed without being timestamped (e.g., an internal approval where prior-existence is not disputed). Most compliance-critical records should have both: a timestamp at the point of data entry (proving the data existed before the analysis) and a signature at the point of review (proving an authorized person approved the record).

7.2 Electronic Signatures — Three Levels for Different Needs

Level 1: Simple (Implicit) Signature

The simplest signature in ELabELN is the act of clicking 'Sign' after re-entering your credentials. No separate PDF is generated. The system records your identity, the timestamp of the signature event, and optionally a brief signature meaning ('I certify that this record is accurate and complete'). This signature level is appropriate for internal review workflows, lab notebooks in academic and industrial research, and any context where the signature's function is attestation rather than legal non-repudiation.

Level 2: Handwritten Signature Block (PDF)

ELabELN can generate a PDF version of the experiment record with a rendered signature block. If the signing user has uploaded a handwritten signature image (in Account Settings), it appears in this block alongside their name, title, and the signature timestamp. This PDF is suitable for situations where a human-readable signed document must be provided to a third party: regulatory submissions, client reports, or inter-institutional sharing.

Note

The handwritten signature image in the PDF is a visual element, not a cryptographic guarantee. The cryptographic record of the signing event is stored in ELabELN's audit log. The PDF is a human-readable representation of that event.

Level 3: Advanced Cryptographic Signature (Ed25519)

ELabELN supports Ed25519 public key cryptographic signatures. When a user signs with this method, the system generates a SHA-256 hash of the experiment's content at the moment of signing, signs that hash with the user's private key, and stores the signature together with the content hash. The signature can be independently verified at any future date: if the experiment content has been altered since signing, the verification fails.

Why it works this way

Ed25519 signatures provide non-repudiation that no password-based signature system can offer. A password can be shared, guessed, or disclosed. An Ed25519 signature can only be produced by the holder of the specific private key, which never leaves the signer's device. For laboratory notebooks used in patent disputes, FDA submissions, or litigation, this cryptographic guarantee is the highest available standard of signature authenticity.

Example: Signature workflow in a pharma QA lab

(1) An analyst completes a method validation study and saves all raw data, calculations, and results in the experiment record. (2) She clicks Sign, re-enters her password and MFA code, selects 'Simple signature' with meaning 'I certify that this record is complete and accurate.' The system logs the signature event. (3) She clicks Request Action and asks her supervisor to review and sign. (4) The supervisor opens the experiment in view mode, reads the record, adds a comment requesting one clarification, receives the response, and then signs with meaning 'Technical review complete. Results are scientifically valid.' (5) Both signatures, with their timestamps and identity records, are part of the permanent audit trail.

7.3 RFC 3161 Trusted Timestamping

RFC 3161 is an international standard for trusted timestamps. The process works as follows:

1. You click Timestamp on an experiment.
2. ELabELN computes a cryptographic hash (SHA-256) of the experiment's complete content at that exact moment.
3. This hash — not the experiment content itself — is sent to an RFC 3161-compliant Timestamp Authority (TSA).
4. The TSA signs the hash with its own private key and returns a timestamp token that binds the hash to the TSA's certified clock.
5. ELabELN stores the timestamp token with the experiment record.
6. At any future date, a reviewer can verify: (a) the content hash matches the stored hash — proving the content has not changed; (b) the TSA's signature is valid — proving the hash was submitted to the TSA; (c) the TSA's certified time is accurate — proving the content existed no later than the timestamp date.

Why it works this way

The critical feature of RFC 3161 timestamps is that the timestamp is issued by an independent, trusted third party that has no relationship with your organization. This independence is what makes the timestamp credible in a dispute. If your own systems generated the timestamp, a skeptic could argue that you set your system clock backwards. A TSA's timestamp is operationally indistinguishable from a postal date stamp: it provides independent, third-party evidence of prior existence.

Example: RFC 3161 timestamping for IP protection

A researcher at a neurotechnology startup develops a novel electrode impedance matching algorithm. She documents the algorithm, its derivation, and initial validation data in an ELabELN experiment. She immediately applies an RFC 3161 timestamp. Seven months later, when a competitor files a patent on a similar algorithm, her attorney can demonstrate that the timestamp token — issued by an independent TSA at a certified time — proves her client's documentation of the method preceded the competitor's filing date. The timestamp is not admissible because it is in ELabELN; it is admissible because an independent, trusted third party certified the date.

7.4 When to Timestamp

Scenario	When to Apply	Why
Novel discovery or invention	Immediately upon documentation	Establishes prior existence for IP purposes. The timestamp is most valuable if applied before the work is shared with anyone outside the core team.
Methodology or protocol development	When the protocol reaches a stable, reproducible form	Protects the methodology as original work and documents the point at which the method was considered validated.
Data set completion	After the final data point is recorded	Proves the data set was complete on a specific date, preventing retroactive addition. Critical for regulatory submissions.
Before any external disclosure	Before submitting to a journal, presenting at a conference, or sharing with a collaborator	The pre-disclosure timestamp establishes a date of record that predates any public or semi-public disclosure.
Routine GMP records	After each analytical run, per SOP	In GMP contexts, timestamping as a routine step (not just for novel work) creates consistent, defense-grade documentation for all regulated records.

8. Audit Trail and Data Integrity

Data integrity is the assurance that a record is complete, consistent, accurate, and trustworthy for its entire lifetime. ELabELN builds data integrity into every layer of the system rather than treating it as a module to be bolted on. This chapter explains the specific mechanisms ELabELN uses and how they map to established data integrity standards.

8.1 Changelog — Field-Level Change History

Every change to a structured field in an experiment (title, date, status, category, custom ID, metadata values, permissions, tags) is recorded in the Changelog. Each changelog entry shows: which field changed, the previous value, the new value, the date and time of the change, and the user who made the change.

Access the Changelog via the Ellipsis (…) menu in the experiment toolbar and selecting 'See Changelog'.

Example: Changelog in action — status change investigation

An auditor reviewing a QA record notices that the experiment status changed from 'Failure' to 'Success' after the analysis was complete. In the Changelog, she sees: 'Status changed from Failure to Success on 2026-04-15 at 14:32 by user jsmith.' She opens the experiment's revisions to see what body text changed at the same time. This investigation — which would be impossible in a paper system — takes 2 minutes in ELabELN and provides a complete, irrefutable record of what changed, when, and by whom.

Why it works this way

Field-level change history is not primarily about catching misconduct. It is about enabling legitimate corrections while maintaining a complete record of those corrections. A researcher who realizes they entered the wrong incubation temperature can correct it — the correction is recorded in the Changelog, the original value is preserved, and the record remains accurate and complete. In a paper notebook, this would require a crossed-out value and initialing, which is difficult to enforce consistently. ELabELN enforces it automatically.

8.2 Revisions — Main Text Version History

The main body of an experiment is versioned. Every time the experiment is saved, a new revision is created if the body content has changed. Access the revision history via the Ellipsis (…) menu and selecting 'See Revisions'.

The revision view shows a list of all saved versions in reverse chronological order, each identified by its save timestamp and the user who saved it. Clicking any revision opens the full text of the experiment body at that point in time. A diff view (side-by-side comparison) is available to highlight exactly what changed between any two revisions.

Why it works this way

The revision system makes the body text equally trustworthy as the structured fields. Without revision history, a researcher could rewrite their experimental narrative after seeing the results,

creating a record that is internally consistent but historically inaccurate. With revision history, the original account of what happened is always recoverable, and every subsequent change to the narrative is documented. This is the digital equivalent of the paper rule 'never use correction fluid' — it is built into the system rather than depending on researcher discipline.

Example: Revision history in a patent dispute

A patent examiner challenges whether a specific synthesis procedure was documented before a competitor's filing date. The researcher's attorney produces the ELabELN revision history showing: (1) the initial entry was created on a specific date; (2) the RFC 3161 timestamp token associated with that revision proves the content existed on that date; (3) subsequent revisions are visible and post-date the timestamp. The revision history combined with the timestamp creates an airtight chain of evidence that the specific content existed at the certified date.

8.3 ELabID — The Permanent Unique Identifier

Every experiment and resource in ELabELN is assigned an ELabID at creation. The ELabID is a system-generated, sequential, permanent identifier that cannot be changed or reassigned. Once an experiment is created, its ELabID is its permanent address in the system.

The ELabID appears in the experiment header, in the URL, in exports, in audit log entries, and in any references or links pointing to the record. Format: a numeric string (e.g., #00412).

Why it works this way

The ELabID exists separately from the user-assigned Custom ID because users make mistakes with manual identifiers: they are duplicated, misspelled, or omitted. The ELabID cannot be duplicated, misspelled, or omitted — it is system-generated and appears automatically. In an audit, if a reviewer references an experiment by ELabID, they can be certain they are looking at exactly the right record. No ambiguity is possible.

Tip

In lab meetings, reference experiments by ELabID rather than by informal names. 'Experiment #00412' is unambiguous; 'the gel from last Tuesday' is not. ELabIDs can be cited in presentations, emails, and reports as permanent, resolvable references to the underlying record.

8.4 Locking, Archiving, and the Deletion Model

State	What It Means and When to Use It
Locking	Prevents further editing of an experiment. The lock event is logged with a timestamp. Only the owner or an Admin can unlock. Unlocking creates a new audit event. Locking is appropriate after signing or after data entry is complete and verified.
Archiving	Moves an experiment out of the active index without deleting it. Archived experiments are accessible via a dedicated archived view and remain fully searchable and retrievable. Archiving is appropriate

	for completed projects, superseded records, or experiments that are no longer relevant to current work.
Soft delete (Recycle bin)	When an experiment is deleted, it goes to the Recycle bin, not immediate permanent deletion. The recycle bin allows recovery of accidentally deleted records. Admins can permanently delete records from the recycle bin. This two-step deletion model protects against accidental loss.
Permanent deletion	Only available from the recycle bin, by Admins only. Should be governed by a formal data retention policy. ELabELN does not auto-purge records.

Use-Case Flexibility

Academic: Experiments are locked after the data is confirmed and a manuscript is submitted. Archived after the manuscript is published. Never deleted (for reproducibility and IP reasons).

GMP pharma: Experiments are locked and signed at the close of each analytical run. Archive policy follows the product's retention schedule (typically 5–10 years post-product lifecycle end). Deletion requires QA approval.

Forensic: Records are locked after the technical review signature. Archives are maintained per jurisdiction retention law (often 25+ years). Deletion of forensic evidence-linked records requires formal authorization.

8.5 ALCOA+ Data Integrity Framework — How ELabELN Satisfies Each Principle

ALCOA+ is the internationally recognized data integrity framework in regulated industries (FDA, WHO, EMA, MHRA). The nine principles are: Attributable, Legible, Contemporaneous, Original, Accurate, Complete, Consistent, Enduring, and Available. ELabELN's architecture is designed to make ALCOA+ compliance structurally impossible to violate during normal use.

ALCOA+ Principle	How ELabELN Satisfies It	Where in ELabELN
Attributable	Every action (create, edit, sign, comment, timestamp, lock) is logged with the user's identity and timestamp. It is technically impossible to create an action in ELabELN without a traceable identity.	All entries, all audit events, all signature records
Legible	Browser-based text renders identically for all readers. No handwriting ambiguity. PDF exports render consistently.	All experiment body text and metadata
Contemporaneous	Auto-save records changes at the moment they occur. Step	Step completion logs, creation timestamps, auto-save revision history

	timestamps record the exact moment of completion. Entry creation timestamps are immutable.	
Original	File attachments are stored immutably with upload timestamps. Original raw data files can be attached at the moment of instrument output. Revision history preserves the original text.	Attachment storage, revision history
Accurate	Metadata fields with defined types and validation reduce transcription errors. Select fields prevent invalid entries. Number fields enforce numeric input.	All custom metadata fields
Complete	Template-defined required fields prevent experiment completion without all necessary data. Step checklists document all protocol steps.	Template metadata fields, step checklists
Consistent	System clock timestamps are consistent and cannot be manually altered. Team-wide templates enforce consistent field definitions across all users.	All timestamps, all template-based experiments
Enduring	Records persist until explicitly deleted (two-step). Archived records remain searchable. LabLynx's managed hosting includes backup and disaster recovery.	Archive system, managed hosting infrastructure
Available	Browser-based access from any device. PDF export for offline or external sharing. REST API for programmatic retrieval. Search enables rapid retrieval of any record.	All access methods, export features, search

9. Collaboration and Sharing

Science is collaborative. ELabELN provides structured tools for sharing work within teams, routing records through formal review processes, and enabling access for collaborators outside the immediate team — while maintaining the integrity, attribution, and auditability of every shared interaction.

9.1 Sharing Within the Team

Within a team, sharing is controlled by each experiment's permission setting (described in Section 3.3). The default for new experiments depends on the team's configuration; most teams default to private (owner only) with researchers opting in to share specific entries.

Team Admins can view all experiments in their team regardless of individual permission settings. This enables supervisors to review student or staff work without requiring each researcher to manually share every entry. It does not override permission settings for non-Admin users: a private experiment is still invisible to non-Admin peers.

> **Example: Team sharing in an academic lab**
>
> A PhD student conducts an experiment and sets its permission to 'User' (readable by all team members). Her PI, who is the team Admin, can see it automatically. Her labmates can also view it (but not edit it). When she is ready for feedback, she uses the comment function rather than changing permissions — the record is already readable. When she wants her labmate to contribute additional data from a follow-up experiment, she changes permission to 'Team' (read-write for all) and leaves a comment requesting the specific contribution.

9.2 Request Action — Structured Review and Approval Workflows

The Request Action feature sends a formal, trackable notification to any ELabELN user requesting them to perform a specific action on an experiment. It converts the informal 'can you please sign my experiment' email into a system-documented workflow event.

Action Type	What It Requests
Archive	Request the recipient to archive this experiment. Useful at project completion when the PI or lab manager reviews and formally closes experiments.
Lock	Request the recipient to lock this experiment. Used when a different person than the owner has the authority to lock (e.g., a QA manager locks all QA-reviewed records).
Review	Request the recipient to review the experiment and provide feedback. The reviewer is notified, reviews in view mode, and responds via comment. The request and all responses are logged.
Sign	Request the recipient to sign the experiment. The most common workflow: analyst completes, requests supervisor signature. The signature, when applied, is logged with full audit trail.

Timestamp	Request the recipient to apply a timestamp. Used when the person with timestamping authority is different from the data entry person.

Example: Multi-step Request Action workflow in a QA lab

Step 1: Analyst completes a stability study experiment, attaches all raw data files, fills all metadata fields, and saves.

Step 2: Analyst requests 'Review' from the Senior Analyst. Senior Analyst reviews the experiment, adds a comment requesting clarification on sample preparation conditions. Analyst responds with the clarification and updates the experiment body.

Step 3: Analyst requests 'Sign' from the Senior Analyst. Senior Analyst signs with meaning 'Peer technical review complete.'

Step 4: Analyst requests 'Sign' from the QA Manager with note 'Please review and approve for inclusion in stability report.' QA Manager reviews, is satisfied, and signs with meaning 'QA approved — suitable for regulatory submission.'

Step 5: QA Manager requests 'Lock' on the experiment. The record is now locked, signed twice, and fully documented. The complete workflow — every request, every response, every signature — is in the audit trail.

Why it works this way

Email-based review workflows are the silent enemy of data integrity audits. They are irretrievable (deleted emails), unstructured (no guarantee the reviewer saw the final version), external to the record (the review is not attached to what was reviewed), and undatable by anyone not on the email thread. Request Action moves the review and approval conversation inside the ELabELN record, where it is part of the same audit trail as the record itself.

9.3 Cross-Team and External Collaboration

Experiments set to 'Organization' permission level are visible to all teams in the ELabELN instance. This enables cross-departmental reference records, shared method libraries, and institutional SOPs accessible to all users.

For external collaborators (researchers at a partner institution, CRO, or regulatory reviewer), ELabELN supports two models:

- Guest account: An external user is granted a 'Locked' account (read-only) for the duration of the collaboration. They can view experiments shared with them but cannot create or edit anything. The access is revoked when the collaboration ends.
- PDF export: For external parties who should not have system access, experiments are exported to PDF and shared as traditional documents. The PDF includes a complete record of all metadata, body text, steps, attachments list, and audit events.

Tip

When creating a guest account for an external reviewer, create a named account (rather than a generic 'auditor' account) so that any access events in the audit log are attributed to a specific individual. Shared accounts make audit log attribution meaningless.

9.4 Transferring Ownership

Ownership of an experiment can be transferred from one user to another via the Ellipsis (…) menu. This is important when a researcher leaves the organization, a project is handed off, or a student graduates and their work is transferred to the PI's account for long-term stewardship.

After transfer, the new owner has full edit rights. The original owner's identity remains in the audit trail and creation records — it is not erased. The transfer event itself is logged.

Why it works this way

In a paper notebook, when a researcher leaves, the notebook often leaves with them or becomes inaccessible. The institution loses both the data and the context needed to interpret it. ELabELN's ownership transfer model ensures that all experiments remain institutionally owned and accessible, regardless of personnel changes. This is a direct answer to one of the most common causes of irreproducible research: data that lives only in one person's files and cannot be found after they leave.

10. Import, Export, and the REST API

ELabELN is designed as an open system: data can be exported in multiple formats, imported in bulk, and accessed programmatically via a comprehensive REST API. This openness is fundamental to ELabELN's role in the laboratory data ecosystem — it is a record-keeping hub, not a data silo.

10.1 Exporting Experiments — Four Formats for Different Purposes

Format	Description and Use Cases
PDF	A fully formatted, human-readable document containing all experiment fields, body text (including images and tables), metadata, steps (with completion timestamps), attachments list, and audit summary. PDF exports are suitable for regulatory submissions, external sharing, printing, and archiving alongside raw data files.
ZIP archive	All files associated with the experiment: the PDF, all attachments in their original format, and a JSON metadata file. The ZIP is a complete, self-contained archive of everything in the experiment record. Suitable for long-term archiving, offline backup, or transfer to another institution.
ELN format (RO-Crate)	A machine-readable archive format based on the Research Object Crate (RO-Crate) specification. This format is designed for interoperability: ELN-format exports can be imported into any ELN system that supports the specification. It is the standard format for laboratory data exchange between institutions.
CSV / JSON (API)	Individual fields and metadata can be exported programmatically via the REST API in JSON format. Custom export scripts can be written to extract specific data sets for analysis, reporting, or LIMS integration.

Tip

For regulatory submissions, always use the PDF export. It is self-contained, human-readable, and includes the audit summary. For archival alongside raw data, use the ZIP archive: it preserves all attached files in their original format alongside the experiment record.

10.2 Importing Resources in Bulk

Resources can be imported in bulk using a CSV upload. This is the fastest way to populate a new ELabELN instance with an existing inventory: export your inventory from a spreadsheet or LIMS, format it to the ELabELN CSV template, and import. Hundreds or thousands of resource records can be created in a single operation.

The CSV template is available from the Resource import page. Required columns are Title and Category. All custom metadata fields for the target category can be included as additional columns.

Example: Bulk import in an academic lab

A new lab has been running on paper for two years. They have a 400-row spreadsheet of antibodies, chemicals, and equipment. They export the spreadsheet to three CSV files (one per resource category), map the column names to the ELabELN field names, and import. All 400 resources are in ELabELN in 20 minutes. The team then begins linking these existing resources to new experiments from day one, rather than spending weeks manually re-entering inventory.

10.3 The REST API — Connecting ELabELN to the Laboratory Ecosystem

ELabELN exposes a comprehensive REST API that enables programmatic read and write access to experiments, resources, templates, and the user directory. Every action possible in the user interface can be performed via the API. Authentication uses personal API keys (generated in Account Settings).

The API documentation is available within your ELabELN instance under the API link in the user menu. It is generated from the live API and is always current with the installed version.

Integration Pattern	How It Works	Applicable Lab Types
Instrument integration	An instrument controller scripts create a new ELabELN experiment automatically after each run, populates metadata fields from the run log, and attaches the raw data file — all without the analyst touching ELabELN	LC-MS systems, plate readers, sequencers, chromatography controllers, spectrophotometers
LIMS synchronization	A LIMS system that manages sample receipt creates a corresponding ELabELN experiment for each sample, pre-populating the Custom ID with the LIMS accession number and the metadata with sample type and origin	High-volume analytical labs, clinical testing, environmental testing
Automated reporting	An overnight script queries ELabELN for all experiments completed in the past 24 hours, extracts result metadata, and populates a daily QC dashboard	QA labs, testing labs, production monitoring
Data analysis pipeline	An R or Python script reads result data from ELabELN experiments, performs statistical analysis, generates a figure, and posts the figure back as an	Bioinformatics, genomics, computational research, high-throughput screening

	attachment to a summary experiment	
Compliance reporting	A monthly script queries all experiments with specific metadata values (e.g., all GMP runs in a product line) and generates a formatted compliance report	Pharma QA, regulatory submissions, stability programs

Why it works this way

An ELN that cannot connect to the rest of the laboratory's data ecosystem creates a new data silo rather than eliminating old ones. The REST API is the answer to the question 'can ELabELN talk to our [instrument / LIMS / ERP / analysis system]?' The answer is yes, for any system that can make HTTP requests — which includes virtually every modern laboratory instrument controller and informatics platform.

11. Compounds, Containers, and Special Features

11.1 Chemical Compounds Database

ELabELN includes a dedicated Compounds module separate from the general resource system. The Compounds module is specifically designed for chemical entities: it includes a molecular structure editor (using the Ketcher drawing tool), automatically computes molecular weight, molecular formula, and SMILES notation from the drawn structure, and integrates with GHS hazard classification.

Feature	Description
Structure editor	Draw or paste chemical structures using the Ketcher structure editor. Supports SMILES, InChI, and MOL file import.
Auto-computed properties	Molecular weight, molecular formula, exact mass, and SMILES are computed automatically from the drawn structure.
GHS integration	GHS hazard pictograms, signal words, and H/P statements can be assigned to each compound. This information appears whenever the compound is linked to an experiment, providing a safety reminder at the point of use.
Compound linking	Compounds can be linked to experiments exactly like resources. All experiments using a specific compound are immediately retrievable from the compound's record.

Use-Case Flexibility

Medicinal chemistry: Every synthesized compound is entered in the Compounds module with its structure. All characterization experiments (NMR, MS, HPLC purity, biological activity) are linked to the compound record. The compound record is the navigable hub for the entire characterization campaign.

Process chemistry: Reagents and intermediates are tracked in Compounds. The GHS data ensures safety information is visible when any researcher links the compound to a new synthesis experiment.

Academic: Students entering new compounds from synthesis experiments use the structure editor, which validates the structure and computes properties automatically, eliminating transcription errors in molecular weight and formula.

11.2 Containers — Hierarchical Physical Storage Tracking

The Containers module tracks where physical materials are stored. Containers nest hierarchically: a Freezer contains Racks; a Rack contains Boxes; a Box contains Slots. Each slot can hold a resource or sample. The entire physical storage hierarchy is represented in ELabELN and navigable by location.

Example: Container hierarchy in a biobank

Level 1 — Freezer: 'Ultra-Low Freezer ULT-01, -80°C, Building B, Room 204'

Level 2 — Rack: 'Rack 3, Shelf 2'

Level 3 — Box: 'Box 14 — Anti-EGFR Antibodies'
Level 4 — Slot: 'Slot C4 — Anti-EGFR mAb Lot 0012, 100 µL aliquot x 5'
The resource record for this antibody includes its container location. When any researcher opens the antibody resource, the location is immediately visible. When the slot is empty, the container record is updated and the resource's location field changes. No separate inventory spreadsheet is needed.

Why it works this way

Container tracking solves the 'where is it?' problem that plagues large laboratories. In a lab with hundreds of antibodies, cell lines, and chemical stocks across multiple freezers and refrigerators, a researcher who does not know an item's location must either search physically or ask a colleague. ELabELN's container module makes the location a searchable, linked property of the resource record. 'Where is the Bradford reagent?' becomes a 2-second search rather than a 10-minute physical hunt.

11.3 QR Code Labels

Every experiment and resource in ELabELN can generate a QR code. The QR code encodes the direct URL to that specific entry. Printed QR code labels can be affixed to physical samples, containers, equipment, or tube racks. Scanning the code with any smartphone or barcode scanner opens the corresponding ELabELN record instantly in the browser.

ELabELN supports label printing directly from the interface: select a label size and print format compatible with Avery or similar label sheets.

Example: QR codes in an environmental testing lab

Sample bottles arrive from field collection with printed labels. Each label carries a QR code that the field technician generated in ELabELN before departure, pointing to the pre-created collection experiment for that sampling event. In the lab, a technician scans each bottle's QR code with a tablet, opening the corresponding field collection experiment. They attach the field technician's metadata and begin the laboratory analysis experiment, linking it to the collection experiment. The chain from field to lab is complete and requires no manual transcription of sample IDs.

Use-Case Flexibility

Cell line management: QR codes on cryovials link to the cell line resource record. Researchers scan the vial before use to confirm the passage number, mycoplasma status, and thaw history.

Forensic evidence: Evidence containers carry QR codes. Every time evidence is accessed, the researcher scans the QR code, opening the evidence resource record and creating a trackable interaction log.

Equipment: Instruments carry QR codes linking to their resource record (with calibration dates and booking calendar). Any researcher can scan the instrument and immediately see whether its calibration is current.

11.4 The Draw Tool and JSON Editor

ELabELN includes a browser-based drawing tool for creating figures, plate maps, lab diagrams, and experimental schematics directly within an experiment record. Drawings are stored as SVG images and embedded in the experiment body. They are scalable, searchable by their text content, and do not require external software.

Common uses for the draw tool include: 96-well plate maps (with sample positions annotated), experimental flow diagrams, molecular cloning schematics, equipment setup diagrams, and gel lane maps. A plate map created in the draw tool next to the assay protocol in the same experiment body is immediately more useful than a separate PowerPoint file.

The JSON editor is an advanced feature that allows direct editing of an experiment's underlying data structure. It is used primarily by Admins for bulk metadata updates, data migration, and troubleshooting. Standard users do not need to interact with it.

12. Configuration Patterns by Laboratory Type

ELabELN's flexibility means it can be configured very differently for different types of laboratories. This chapter presents five detailed configuration patterns: a recommended starting setup for each lab type, with specific guidance on categories, templates, metadata fields, permission settings, workflows, and features to enable. These patterns are starting points, not prescriptions.

12.1 Academic Research Lab

Goals: Reproducibility, collaboration, grant compliance, IP documentation, publication support, student training.

Configuration Area	Recommended Setup
Categories (start here)	Cloning & Molecular Biology, Protein Expression, Cell Culture, Biochemical Assays, Animal Experiments, Computational Analysis, Data Analysis & Statistics, Literature & Protocol Design
Template priority	Create templates for your 5–10 most frequently repeated procedures first. Use the Bradford assay, your standard Western blot, and your most common cell viability assay as templates. Novel experiments start blank.
Metadata fields to always include	Experiment Objective (Text area), Hypothesis (Text area), Conclusion (Text area), Related Publication / Grant Code (Text), Key Result (Radio: Positive / Negative / Inconclusive)
Permission default	Private (owner only). Team members can view after the researcher shares. PI (Admin) can always see all entries.
Signature workflow	Researcher signs when the experiment is complete and data is confirmed. PI signs for grant-critical or IP-sensitive experiments. RFC 3161 timestamp applied to novel discoveries.
Collaboration features	All publications: link all contributing experiments to a 'Manuscript — [Title]' summary experiment. Student handoff: transfer ownership of all experiments to PI account at graduation.
Training use	New students are given User accounts and assigned to work through a series of template-based experiments. The PI can review their work in real time via Admin access.

12.2 Pharmaceutical / Biotech Quality Assurance Lab

Goals: 21 CFR Part 11 compliance, CAPA documentation, method validation, batch release testing, audit readiness.

Configuration Area	Recommended Setup

Categories	Release Testing, Stability Study, Method Validation, In-Process Control, CAPA Investigation, Reference Standard Qualification, Equipment Qualification
Templates	All routine methods must have a locked-step template. Template version is a required metadata field. Method templates are reviewed and approved by QA before release.
Required metadata fields	Analyst (Select from trained personnel list), Supervisor (Select), Instrument (Select from calibrated instruments), Method Number (Text), Lot Number (Text), Batch ID (Text), Pass/Fail (Radio)
MFA	Required for all user accounts. Enforced at the system level.
Signature workflow	Analyst signs upon completion. Supervisor/QA signs upon review. Both signatures use credential re-entry. Ed25519 signatures for regulatory submissions.
Audit readiness	All experiments locked and signed before month-end. Status 'Pending QA Review' used to queue records for supervisor sign-off. Audit exports available as PDF or ZIP per experiment.
CAPA integration	Each CAPA investigation is an experiment in the 'CAPA Investigation' category. It links to the OOS result experiment that triggered it. The CAPA record documents investigation, root cause, corrective action, and effectiveness check.

12.3 Forensic / Crime Laboratory

Goals: Chain of custody integrity, evidence traceability, court-ready records, analyst competency documentation.

Configuration Area	Recommended Setup
Categories	Drug Chemistry, DNA Analysis, Toxicology, Trace Evidence, Firearms, Digital Evidence, Questioned Documents, QC / Proficiency Testing
Custom ID usage	Case Number is the Custom ID for all evidence-related experiments. All experiments for one case carry the same Custom ID. A search by case number returns the complete analytical record for that case.
Required metadata fields	Case Number (Text), Item Number (Text), Analyst (Select), Technical Reviewer (Select), Instrument (Select), Analysis Method (Select from accredited methods), Disposition (Radio: Positive/Negative/Inconclusive/Inconclusive-Insufficient Sample)
Signature workflow	Analyst signs upon completion of analysis. Technical reviewer signs as part of the mandatory technical review. Both signatures are required before a result can be included in a report.
Evidence resources	Physical evidence items are resource records. The evidence resource record tracks: description, chain of custody log, storage location, and all analyses performed (linked experiments).

QC and proficiency	QC and proficiency test experiments are in a dedicated category, reviewed by the quality manager, and retained separately for accreditation audit purposes.
Permissions	All case-related experiments are private to the assigned analyst and reviewers. No case data is visible to analysts on other cases.

12.4 Neurotechnology / Medical Device R&D Startup

Goals: Rapid deployment, IP protection, multi-team coordination, regulatory readiness, investor documentation.

Configuration Area	Recommended Setup
Team structure	Teams per functional area: Hardware Engineering, Firmware, Neuroscience Research, Clinical Preclinical, Regulatory Affairs. Cross-team sharing via Organization-level experiments for shared protocols.
Categories per team	Neuroscience: Animal Experiments, In Vitro Experiments, Data Analysis, Equipment Validation. Hardware: Device Fabrication, Impedance Characterization, Bench Testing, Failure Analysis.
IP protection workflow	Novel discovery or invention: RFC 3161 timestamp applied immediately upon documentation. Ed25519 signature by the inventor. The timestamped, signed record is cited in the patent application as prior art evidence.
Instrument API integration	Acquisition systems write recording metadata directly to ELabELN via REST API after each session. The analyst opens the pre-populated experiment and adds their analysis. No manual transcription of session parameters.
Regulatory readiness	21 CFR Part 11 compliant features (MFA, audit trail, electronic signatures) are enabled from day one, even before regulatory engagement. This avoids the expensive retroactive remediation that many startups face at the pre-submission meeting.
Scale	Start with 3–5 users and unlimited-user pricing. Add team members as the organization grows. No per-seat cost, no migration required.

12.5 ISO 17025 Analytical Testing / Quality Laboratory

Goals: Traceable method execution, equipment management, QC documentation, proficiency testing records, audit readiness.

Configuration Area	Recommended Setup
Categories	Routine Analysis (per method type), Method Development, Method Verification, QC Check, Equipment Calibration, Proficiency Testing, Corrective Action, Uncertainty Estimation

Template structure	One template per accredited method. All steps locked. Method version is a required metadata field. Template is reviewed and approved by the quality manager before release. Old template versions are archived, not deleted.
Equipment management	All instruments are resource records with: calibration due date (Date field), calibration certificate number (Text), maintenance log (body text, updated per event), and bookable calendar. Before any analysis, the analyst checks the instrument resource to confirm current calibration covers the analysis date.
QC requirements	QC check experiments are linked to the routine analysis experiments for which they verify instrument performance. QC data is tracked in a dedicated 'QC Check' category with threshold metadata fields and Pass/Fail radio field.
Proficiency testing	Each proficiency test event is an experiment. Results are entered, linked to the relevant method verification experiments, and signed by the quality manager. The record is maintained for accreditation review.
Audit preparation	Monthly: Admin generates a filtered view of all experiments in the past 30 days, sorted by category and analyst. All should be locked and signed per the QMS requirement. Any unsigned records are flagged and completed before the next audit cycle.

13. Tips, Best Practices, and Common Pitfalls

13.1 Daily Use Habits That Make a Difference

Habit	Why It Matters
Document before you analyze	Create and populate the experiment record before you open your results in a spreadsheet or analysis software. The discipline of documenting first prevents the unconscious bias of shaping the record around a result you already know.
Attach raw data immediately	As soon as an instrument produces output, attach the file to the corresponding experiment. Do not batch this task to the end of the day. Files that are not attached immediately get lost, renamed, or forgotten.
Use descriptive titles	Spend 30 seconds writing a title that describes the experiment's content: technique, key variable, outcome or purpose. This investment compounds over time as your records become self-findable.
Check off steps as you go	If your experiment has a step checklist, check off each step at the moment of completion — not all at once at the end of the procedure. The timestamp value of a step that was checked off at the end of the day is meaningless. The timestamp value of a step checked off when it occurred is evidence.
Comment generously	Add comments to your own entries when you notice something unexpected, when you make an unplanned deviation, or when you have a retrospective insight. Future you, your collaborators, and your reviewers will all benefit. Comments cost 30 seconds; reconstructing undocumented observations can take days.
Link as you work	When you use a reagent, link the experiment to the reagent's resource record. When you build on previous work, link the new experiment to the earlier one. Links are trivially easy to create and enormously valuable to follow later.

13.2 Common Pitfalls and How to Avoid Them

Pitfall	Problem It Creates	How to Avoid It
Copying text from external files	Body text pasted from Word or PDF retains invisible formatting markup that can corrupt the experiment record over time	Paste as plain text (Ctrl+Shift+V) or use the 'Paste without formatting' option in the editor toolbar
Inconsistent tag names	'Western Blot', 'western blot', 'WB', and 'Western-blot' are four different tags. Over	Establish and publish a team tag convention before data accumulates. Use the tag

	time, tagged experiments become unretrievable because the tag is inconsistent	autocomplete to pick existing tags rather than typing new ones.
Generic titles	'Experiment 1' or 'Test run' are unsearchable and meaningless to anyone but the creator	Write titles that include technique, variable, and purpose. See Section 3.3 for examples.
Attaching files at end of project	Batch-attaching files weeks or months after the experiment occurred decouples the file from its context and creates attachment timestamps that do not reflect when the data was collected	Attach files at the moment of data generation. Make it part of the procedure, not a documentation task.
Ignoring the resource module	Creating experiments without linking to resources means there is no connection between experiments that used the same reagent lot. Impact analysis becomes impossible.	Create resource records for your top 20–30 most-used items and link all new experiments to them. Expand resource coverage incrementally.
Sharing access too broadly	Setting all experiments to Organization permission out of convenience exposes unpublished data to the entire instance	Default to private or team-level permission. Share more broadly only when the work is ready for broader visibility.
Using only one tag per experiment	A single tag per experiment collapses multi-dimensional classification into a flat list. Experiments become retrievable by only one dimension.	Apply 3–5 tags per experiment covering different classification dimensions: project, technique, material, collaborator, outcome.

13.3 Five Template Design Principles

7. Design for the most junior user, not the most experienced one. A template that an experienced researcher considers over-specified gives a new team member everything they need without asking. A template that an experienced researcher considers 'just right' may leave critical information gaps for someone less familiar.

8. Include all fields you will ever need for analysis, not just the ones you think you will use. Storage costs nothing. Missing data costs everything. Add fields that seem optional; leave them blank when not applicable rather than rebuilding the template later.

9. Lock only what must not vary. Locking every step in a template turns it into a rigid form that frustrates researchers when they need to adapt. Lock the steps that have regulatory, compliance, or quality force. Leave others flexible.

10. Version your templates. When a protocol changes, create a new template version. Archive the old one. Experiments created from the old template retain their original structure; new experiments use the new version. The version history of your protocols is embedded in the template history.

11. Review templates quarterly. Templates that are not being used are either redundant or poorly designed. Templates that are being used but generating inconsistent data need revision. Assign a template owner (an Admin or experienced researcher) who is responsible for reviewing and maintaining the team's template library.

14. Support and Resources

LabLynx is committed to the success of every ELabELN deployment. Support resources are available at multiple levels, from self-service documentation to direct access to the LabLynx technical team.

Resource	Details
ELabELN Knowledge Base	Searchable self-service documentation covering all features, configuration options, and common workflows. Available at: support.lablynx.com
Email Support	Direct access to the LabLynx technical support team for configuration questions, technical issues, and feature inquiries: support@lablynx.com
Onboarding Consultation	LabLynx provides structured onboarding sessions for new deployments: system configuration review, template design assistance, user training, and workflow setup.
Account Management	Your dedicated LabLynx account manager for strategic discussions, product roadmap questions, and expansion planning: contact via support@lablynx.com.
ELabFTW Community	ELabELN is built on eLabFTW. The active eLabFTW community forum at community.elabftw.net provides peer support, feature discussions, and plugin contributions from the global user base.
Status Page	Real-time system status and incident notifications: status.lablynx.com

LabLynx's Promise

ELabELN is more than software — it is a partnership between your laboratory and LabLynx.

We manage the infrastructure, the security, the backups, and the updates so your team can manage the science. If you have a configuration question, a workflow challenge, or a compliance requirement you are not sure how to address in ELabELN, we are a direct email away.

support@lablynx.com | www.lablynx.com | www.elabeln.com